Weekly Report (2017.8.21-2017.8.27)

TASK	DEADLINE	CURRENT PROGRESS
Visual analysis system of LAN	8.31	Try to use a method that is presented in an existing work to resolve the relationship problem. Now the problem is the drawing of relationship tree and the development of some details. And current proposed solution is that take the time to do it step by step.

Done

1) Paper Reading: Quasi-biclique Edge Concentration: A Visual Analytics Method for Biclustering PVis 2017

Firstly, biclustering is a well-known approach for data mining, and it is applied in many fields, such as genome analyses, security services, and social network analyses. Biclustering finds bicliques contained in a bipartite graph, namely, it finds biclusters from the relationship between two entity sets. Quasi-biclique is a mathematical concept that represents incomplete bicliques.

However, in real data, a biclique may lack several edges because of various reasons, such as errors. In this situation, traditional biclustering methods cannot find correct biclusters. A novel biclustering method that can analyze real data under uncertainty is needed. Thus, this work proposes the quasi-biclique edge concentration (QBEC) method, which is a visual analysis method for biclustering using quasi-biclique mining. QBEC includes visual representations and user interactions for quasi-bicliques. The incompleteness of a quasi-biclique is reflected in edge opacity. Users can interactively explore data by adjusting the incompleteness parameter of the quasi-biclique.

The algorithm of this work can be used in my current LAN project. It employs Liu'definition (Fast Quasi-Biclique Mining with Giraph Liu'algorithm, and the definition if more clear in this work) with a little modify. This work divide the algorithm into 2 parts: pre-process and the main process, shown in Algorithm 1 and Algorithm 2 after.

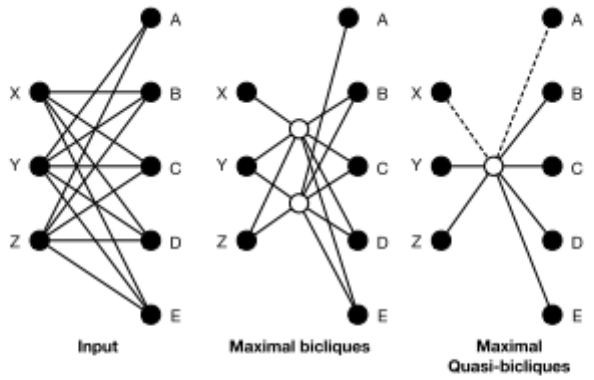

Figure 1: Main concept of the quasi-biclique edge concentration (QBEC). By applying conventional edge concentration, the data is represented as two biclusters. On the other hand, by applying QBEC, the data can be represented as one bicluster.

Algorithm 1: Pre-process for mining μ-quasi-bicliques of $G = (U, L, E)$.

input : A Bipartite graph $G = (U, L, E)$
output : A map that represents candidates of $\mu - quasi - bicliques$C

```
C ← empty map;
for u ∈ U do
    T ← outVertices(u);
    M ← empty map;
    key ← genkey(T);
    C[key] ← (T, M);
end
for key ∈ C.keys() do
    (T, M) ← C[key];
    for v ∈ T do
        for u ∈ inVertices(v) do
            if u ∈ M.keys() then
                M[u] ← M[u] + 1;
            else
                M[u] ← 1;
            end
        end
    end
end
```

Algorithm 2: Main process for mining μ-quasi-bicliques of $G = (U, L, E)$.

input : A bipartite graph $G = (U, L, E)$, threshold μ, and C
output : A set of maximal μ-quasi-bicliques K

```
K ← empty set;
for key ∈ C.keys() do
    (T, M) ← C[key];
    S ← φ;
    for u ∈ M.keys() do
        if M[u] ≥ μ|T| then
            S ← S ∪ {u};
        end
    end
    if |S| > 1 and |T| > 1 then
        K ← K ∪ {(S, T)};
    end
end
filterNonMaximalQBC(K);
```

2) Implement the fisheye distortion to have a look at the details of data.

The introduction of fisheye:

https://github.com/d3/d3-plugins/tree/master/fisheye

The demo of fisheye:

https://bost.ocks.org/mike/fisheye/, I got the source code by pressing the F12.
The implement of applying the distortion to the only X dimension.
http://jsfiddle.net/t0pbqe3j/2/. When this is used in my project, I compute the new position when mousemove event is triggered and redraw the nodes and links based on new position.
But now we still have some questions, Mouse mousemove event is too sensitive, it needs to slow down, or bad to do click events.
3) Store the CSV data into the database one by one.
Mysql use specification: https://github.com/anjuke/coding-style/blob/master/mysql/mysql-guideline.md. In the use of string concatenation of the SQL statement, it is easy to meet an error, such as less brackets, commas, etc., there is not a hyphen -, we should use underscore to connect character instead of hyphen.
When the property name of an object is concatenated by a hyphen, such as a-b-c, in the access to the property, we cannot write the code as follows: obj.a-b-c, this writing can only get the obj.a and an error. The correct wording is: obj ['a-b-c'].
4) Flexible use of link in d3.js. LinkHorizontal is also a kind of link, so the functions of link can be used to linkHorizontal. Such as source() and target(), to draw the connection of two points.

To Do

1) Finish the development of system as fast as I can.